

What's new in version 24?

Custom Dialog Builder

The Custom Dialog Builder gets a major upgrade, including new controls, new properties of existing controls, new options for the menu location of a dialog, and an updated interface.

- You can specify to install a custom dialog to a new submenu that is created when the dialog is installed. You can also specify whether a custom dialog is installed to all window types (Data Editor, Syntax, and Viewer) or to only a particular window type.
- The following controls are now supported: field chooser, dataset selector, date control, secured text control, color picker, table control, and tab control.
- Controls can be enabled or disabled based on the state of another control.
- You can specify the minimum or maximum number of fields for a target list. You can also specify the minimum or maximum number of selected items for a multi-select list box.
- You can specify whether a combo box control is editable. When the control is editable, a custom value can be entered at run time.
- Labels can be positioned to the left of the control for the following controls: combo box, text control, number control, and file browser.
- For controls that generate a list of items, you can specify the delimiter between the items in the generated syntax. This property applies to target lists, field choosers, table controls, and list box controls.
- You can specify the width of the text area for the following control types: text control, number control, and secured text control.
- The number of functional columns on the canvas, in which you can place controls, is increased to four. Previously, only three functional columns were supported.
- You can specify that the content of a text control is a dataset name, which is validated at run time.
- You can specify how to handle interior quotation marks in values that are enclosed in single or double quotation marks. This property applies to the following control types: text control, table control, combo box, and list box.

Programmability enhancements

- Easily find and download new extensions; get updates to existing extensions; remove installed extensions; and view properties of your installed extensions from the new Extension Hub.
- You can now choose to run Python code with either the Python 2 or the Python 3 processor. By default, Python code uses the Python 2 processor.

Reading and writing Excel files

- Determine the data format for a column based on a specified percentage of values with the same format. Previously, if all values in a column did not have the same format, the column was assigned the string format. You can now specify the percentage of values that determine the format. The default setting is 95%.
- Ignore hidden rows and columns.
- Remove leading and trailing spaces from string values.
- Specify a sheet name when writing Excel data files.
- Append a named sheet to an existing Excel workbook.
- Write variable labels instead of variable names to an Excel file.

Reading text data files

- Simple but flexible interface for reading CSV data files.
- More flexibility in reading text data files, including automatic detection of data formats and better handling of date and time values.

Custom Tables

The Custom Tables option includes a number of new features.

- Display significance values for column means and column proportions tests.
- Merge significance test results into the main table.
- Confidence intervals and standard errors for most summary statistics.
- Weighted results based on an effective base sampling weight variable.

Date/time Formats

- You can read and display datetime values in the form yyyy-mm-dd hh:mm:ss with the new `YMDHMS` format. And you can read and display time durations in the form mm:ss with the new `MTIME` format.
- You can omit delimiters (with some limitations) in input values for `DATE`, `ADATE`, `EDATE`, `SDATE`, and `TIME` formats.